

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 1

JC542 U.S. PTO
09/209162
12/10/98

```
1  --Mailbox movie contains the following variables:
2
3  on startMovie
4      global mbxG_username      -- current user name
5      global mbxG_messages     -- list of messages
6      global mbxG_nMsgs        -- number of messages in mailbox
7      global mbxG_boxName      -- current mailbox name
8      global mbxG_whichLine    -- current hilite line = msgNumber
9      global mbxG_subtractLine -- does this read the previous line as
10 well?
11
12      -- global mbxG_clickCount
13      -- may be used for double click on message to pass
14      -- message to API
15      -- used because "if the doubleClick" in Director is buggy, but
16
17      -- called by API to Main Movie
18      tell the stage to emh_continue(#mailbox)
19
20 end
21
22 -----
23 -- StopMovie handler in a MIAW is called only when the movie
24 -- plays through to the end or jumps to another movie.
25 -- It isn't called when the window is closed or the window
26 -- is deleted by the forget window command.
27
28 on stopMovie
29
30     cleanUpMovie()
31
32 end
33
34 -----
35 -- called to close a MIAW or automatically whenever
36 -- forgetWindow is called
37
38 on closeWindow
39
40     cleanUpMovie()
41
42 end
43
44 -----
45 -- cleanUpMovie can be called by both stopMovie and
46 -- closeWindow.
47 clickCount is a global for the doubleclicking function..
48
49 on cleanUpMovie
50     global mbxG_whichLine, mbxG_subtractLine
51     global mbxG_nMsgs, mbxG_username, mbxG_clickCount
52
53     -- CLEAR FIELDS AND GLOBAL VARIABLES
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 2

```
54
55     put "" into field "MailboxTitle"
56     put "" into field "prepositionTitle"
57     put "" into field "MailboxTo"
58     put "" into field "MailboxSubject"
59     put "" into field "MailboxDate"
60
61     set mbxG_whichLine = 0
62     set mbxG_subtractLine = 0
63     set mbxG_nMsgs = 0
64     set mbxG_clickCount = 0
65
66     if findEmpty(member 50) > 50 then
67         set the scriptText of member 50 = ""
68     end if
69
70     -- SET HIGHLIGHT OFF STAGE
71
72     set the loc of sprite 11 to point (11, -17)
73
74 end cleanUpMovie
75
76
77 -- API Public Handlers
78 -----
79
80 --- Ugly hack to work around problem with Director startup
81 --- of MIAWs. The problem is that, after calling a handler in the
82 --- MIAW, the StartMovie handler for the MIAW does not run until
83 --- the calling movie advances to its next frame.
84 --- Therefore, the calling sequence in the calling movie
85 --- has to be engineered so that the real handlers in the MIAW do not
86 --- run until after control has been transfered back to the calling
87 --- movie. However, at least one handler in the MIAW must be called
88 --- by the calling movie before the StartMovie handler will run.
89
90 --- startMeUp is the fake handler that, when called by the
91 --- main movie, will upon return to the main movie,
92 --- cause this movie's startMovie handler to run.
93
94 --- The second part of this wormy hack is contained in the MIAW's
95 --- startMovie handler... It is a call to a workAround handler in
96 --- the calling movie called continueComponent
97 --- The calling movie's continueRebus handler calls the real handlers
98 --- in the MIAW.
99
100 on emc_startMeUp
101     -- put "Macromedia sucks!"
102     return(1)
103 end emc_startMeUp
104
105 -----
106 -----
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 3

```
107 -- initWindow is called by email main when a message handler
108 -- is opened
109
110 on emc_initWindow userName
111     global mbxG_whichLine, mbxG_subtractLine, mbxG_username
112     -- mbxG_whichLine: track user line selected
113     -- mbxG_subtractLine: allow highlight to follow user selection
114     -- mbxG_msgNum: tracks user selected message number
115
116     set mbxG_username = userName
117     set mbxG_whichLine = 0
118     set mbxG_subtractLine = 0
119
120     -- PUPPET THE HIGHLIGHT
121
122     puppetSprite 11, TRUE
123
124     -- SET HIGHLIGHT OFF STAGE
125
126     set the loc of sprite 11 to point (11, -17)
127
128     return(1)
129 end emc_initWindow
130
131 -----
132
133 -- more API Public Handlers
134 -----
135
136 --- closeWindow is not called unless Rebus plays as a MIAW.
137
138 on emc_closeWindow
139     closeWindow()
140     return(1)
141 end emc_closeWindow
142
143
144 -----
145
146 on emc_getComponentInfo
147
148     -- eventually the MIMEtype field will be application/txt
149     return( list( "SimpleMail", 1, #mailbox, "" ) )
150
151 end emc_getComponentInfo
152
153
154 -----
155 -- the following was called "mbx_giveMessage" with return(0),
156 -- but was not called from anywhere and not on API sheet, thus
157 -- no longer a part of the API.
158
159 on mbx_getMessage
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 4

```
160
161    -- "open" button and doubleClick of highlighted message (not
162    -- yet implemented) calls to email main to hand the message
163    -- selected to a message handling movie
164    -- This script was previously the "open" cast member script:
165
166    global mbxG_whichLine, mbxG_messages
167
168    set mailData = getAt(mbxG_messages, mbxG_whichLine)
169
170    tell the stage
171        emh_passMessage(mailData, mbxG_whichLine)
172    end tell
173
174 end mbx_getMessage
175
176 -----
177
178 on mbx_getMessageNumbers
179
180     -- fill me in!
181     -- list of int mbx_getMessageNumbers
182
183 end mbx_getMessageNumbers
184
185 -----
186 -- Trash messages should return a list of message numbers that are to
187 -- be trashed in the mailfile. Email main will rewrite the mail file
188 -- When implemented correctly, it will determine which message numbers
189 -- are associated with the currently selected lines in the mailbox
190 -- display, update the display to remove these messages from the
191 -- list, and return the list of deleted message numbers.
192
193 on mbx_trashMessages
194
195     --- needs implementation that can handle multiple messages
196     --- also need to rewrite trashIt which does not conform to
197     --- API rule that only API handlers can be called in other movies
198
199     trashIt()
200     return([])
201
202 end mbx_trashMessages
203
204 -----
205 -----
206 -- accepts a mailbox datastructure that consists of a boxname and
207 -- a list of messages
208
209 on mbx_openMailbox mailbox
210     global mbxG_username, mbxG_messages, mbxG_boxName, mbxG_nMsgs
211
212     set mbxG_boxName = getAt(mailbox, 1)
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 5

```
213     put mbxG_userName & "'s " & mbxG_boxName into field "mailboxTitle"
214
215     set mbxG_messages = getAt(mailbox, 2)
216     set mbxG_nMsgs = count(mbxG_messages)
217
218     displayMailbox(mbxG_messages)
219     return(1)
220
221 end mbx_openMailbox
222
223 -----
224
225 --- score script ss_goTheFrame
226
227 on exitFrame
228
229     go the frame
230
231 end
232
233
234 on formatFields
235
236     -- FORMAT THE TEXT FIELDS
237
238     setTextInfo "MailboxTo", " ", "left", "arial", 14, "bold"
239     setTextInfo "MailboxSubject", " ", "left", "arial", 14, "bold"
240     setTextInfo "MailboxDate", " ", "left", "arial", 14, "bold"
241
242 end formatFields
243
244
245 -- places the appropriate components from each message
246 -- into field members with lines aligned for display
247
248 on displayMailBox msgList
249
250     repeat with msg in msgList
251         put getProp(msg, #from) & RETURN after field "MailboxTo"
252         put getProp(msg, #re) & RETURN after field "mailboxSubject"
253         put getProp(msg, #date) & RETURN after field "mailboxDate"
254     end repeat
255
256 end displayMailbox
257
258 on hiliteMessage
259     global mbxG_nMsgs, mbxG_whichLine, mbxG_subtractLine
260
261     -- KEEP TRACK OF SELECTED LINE
262
263     set mbxG_whichLine = the mouseLine
264
265     -- MAKE SURE LINE IS VALID
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 6

```
266
267   if mbxG_whichLine <= 0 then
268       return(0) -- do nothing, errors are caught elsewhere
269       else if mbxG_whichLine > mbxG_nMsgs then
270           -- user clicked somewhere else in field
271           set mbxG_whichLine = 0 -- reset to 0
272           return(0)
273   end if
274
275   -- HIGHLIGHT SELECTED LINE
276
277   set whichHighlight = mbxG_whichLine + mbxG_subtractLine
278
279   -- since all field members in display are kept synchronized
280   -- any one will do for linePosToLocV
281   -- use "MailboxTo", it's small
282
283   set the locV of Sprite 11 to ~
284       (99 + linePosToLocV(member "MailboxTo", whichHighlight))
285
286   end hiliteMessage
287
288   _____
289
290   --- script of cast member mailboxTo
291
292   on mouseUp
293       hiliteMessage()
294
295   end
296
297   --- script of cast member mailboxSubject
298
299   on mouseUp
300       hiliteMessage()
301
302   end
303
304   --- script of cast member mailboxDate
305
306   on mouseUp
307       hiliteMessage()
308
309   end
310
311   _____
312
313   end
314
315   _____
316
317   _____
318
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 7

```
319 --- script of cast member Up
320
321 on mouseDown
322
323     global mbxG_whichLine, mbxG_subtractLine
324
325     -- SCROLL UP WITH HIGHLIGHT
326
327     repeat while the mouseDown = TRUE
328
329         -- GET NUMBER USED TO CORRECT FOR DISCREPANCY BETWEEN
330         -- THE MOUSELINE AND THE LINEPOSTOLOCV
331         -- THE MOUSELINE GIVES LINE WITHIN FIELD TOTAL
332         -- THE LINEPOSTOLOCV USES LINE OF FIELD ON SCREEN
333
334         set mbxG_subtractLine = mbxG_subtractLine + 1
335
336         if mbxG_subtractLine > 0 then
337             set mbxG_subtractLine = 0
338         end if
339
340         -- SCROLL ALL FIELDS TOGETHER
341
342         scrollByLine member "MailboxTo", -1
343         scrollByLine member "MailboxSubject", -1
344         scrollByLine member "MailboxDate", -1
345
346
347
348         -- MOVE HIGHLIGHT WITH LINE, MOVING HIGHLIGHT
349         -- OFF SCREEN WHEN LINE MOVES OFF SCREEN
350
351         set whichHighlight = mbxG_whichLine + mbxG_subtractLine
352
353         if whichHighlight <= 0 or whichHighlight >= 22 then
354             set the loc of sprite 11 to point (11, -17)
355         else
356             set the locV of Sprite 11 to (99 + linePosToLocV(member
357 "MailboxTo", whichHighlight))
358         end if
359
360         updateStage
361
362     end repeat
363
364 end
365
366 -- script of cast member Down
367
368 on mouseDown
369
370     global mbxG_whichLine, mbxG_subtractLine
371
```

**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 8

```
372      -- SCROLL DOWN WITH HIGHLIGHT
373
374
375      -- SCROLL ALL FIELDS TOGETHER
376
377      repeat while the mouseDown = TRUE
378
379          scrollByLine member "MailboxTo", 1
380          scrollByLine member "MailboxSubject", 1
381          scrollByLine member "MailboxDate", 1
382
383
384          -- GET NUMBER USED TO CORRECT FOR DISCREPANCY BETWEEN
385          -- THE MOUSELINE AND THE LINEPOSTOLOCV
386          -- THE MOUSELINE GIVES LINE WITHIN FIELD TOTAL
387          -- THE LINEPOSTOLOCV USES LINE OF FIELD ON SCREEN
388
389          set mbxG_subtractLine = mbxG_subtractLine - 1
390
391          set amountOfMail = the lineCount of member "MailboxTo"
392
393          if mbxG_subtractLine < - amountOfMail + 1 then
394              set mbxG_subtractLine = - amountOfMail + 1
395          end if
396
397
398          -- MOVE HIGHLIGHT WITH LINE, MOVING HIGHLIGHT
399          -- OFF SCREEN WHEN LINE MOVES OFF SCREEN
400
401          set whichHighlight = mbxG_whichLine + mbxG_subtractLine
402
403          if whichHighlight <= 0 or whichHighlight >= 22 then
404              set the loc of sprite 11 to point (11, -17)
405          else
406              set the locV of Sprite 11 to (99 + linePosToLocV(member
407 "MailboxTo", whichHighlight))
408          end if
409
410          updateStage
411
412      end repeat
413
414  end
415
416  -- script of cast member Open
417
418  on mouseUp
419
420      global mbxG_whichLine, mbxG_messages
421
422      if mbxG_whichLine = 0 then
423          alert "Select a message by clicking with your mouse."
424          return(0)
```


**Appendix B: KidCode® Lingo Client/Server Sample Mailbox
Component Scripts**

Page 9

```
425     end if
426
427     set mailData = getAt(mbxG_messages, mbxG_whichLine)
428
429
430     tell the stage
431         emh_passMessage(mailData, mbxG_whichLine)
432     end tell
433
434 end
435
436
437
438
439
440 -- script of cast member closeWindow
441
442 on mouseUp
443     -- Close the mailbox window
444     tell the stage to emh_killComponent()
445 end
446
447
448
449
450
451
452
453 -- script of cast member mailBoxHighlight
454
455 on mouseUp
456     if the doubleClick then mbx_getMessage()
457     -- Jeff is working on this:
458     -- global mbxG_whichLine, mbxG_subtractLine, mbxG_messages,
459     mbxG_clickCount
460     --
461     -- set whichHighlight = mbxG_whichLine + mbxG_subtractLine
462     -- set mailData = getAt(mbxG_messages, mbxG_whichLine)
463     --
464     -- repeat while whichHighlight <> 0
465     --
466     --     put "true"
467     --     tell the stage
468     --         emh_passMessage(mailData, mbxG_whichLine)
469     --     end tell
470     -- end if
471     -- end repeat
472 end
```